WHAT IS CLAIMED IS:

1. A computer-implemented method for logging messages, comprising:

receiving a configuration request, the configuration request indicating a set of one or more devices to which to log messages;

instantiating the set of one or more devices;

receiving a request to log a message; and

publishing the message to the set of one or more devices.

2. The method of claim 1, wherein publishing the message comprises creating a trace object.

3. The method of claim 2, wherein the trace object is formatted in accordance with an extensible markup language (XML).

4. The method of claim 2, wherein each device is associated with an indication of the types of messages logged by the device.

5. The method of claim 4, wherein publishing the message further comprises sending a pointer to each device that logs messages of a type associated with the message, the pointer pointing to memory that includes the trace object.

6. The method of claim 1, wherein the configuration request is included in a command line parameter associated with an application that requests logging of one or more messages.

7. The method of claim 6, wherein the command line parameter indicates that the set of devices to which to log messages is in a database.

8. The method of claim 7, wherein the database comprises a registry.

9. The method of claim 1, wherein the configuration request is included in an environment variable.

10. The method of claim 1, further comprising calling a filter to indicate that the message is available to be logged.

11. The method of claim 10, wherein the filter is called through a callback function.

12. The method of claim 10, further comprising providing a notification by the filter that a test has completed.

13. The method of claim 10, further comprising processing the message to determine whether to send the message or data derived from the message to a device.

14. The method of claim 1, wherein the request to log a message comes from a current thread.

15. The method of claim 14, wherein publishing the message comprises providing a context identifier that identifies a context of the current thread.

16. The method of claim 15, wherein the context identifier further identifies a context of a parent thread associated with the current thread.

17. The method of claim 1, wherein publishing the message comprises providing information that uniquely identifies the thread.

18. The method of claim 17, wherein the information comprises an identifier that identifies a machine on which the current thread executes, a name of a process that spawned the current thread, an identifier that identifies the process, and an identifier that identifies the thread.

19. The method of claim 1, wherein receiving the configuration request occurs after an application that requests to log the message has been compiled, such that the application is not required to be recompiled to publish messages to the set of one or more devices.

20. The method of claim 1, wherein the message is published on a first machine and wherein the request to log the message is received from a second machine.

21. The method of claim 20, further comprising combining a request to log a first message from the first machine with a

request to log a second message received from the second machine before publishing the message on the first machine.

22. A computer-readable medium having computer-executable instructions for performing the method of claim 1.

23. A system for logging messages, comprising:

a logger having an interface configured to receive a request to log a message;

a local publisher configured to receive a log message from the logger; and

a set of one or more devices configured to log messages published by the local publisher, the set selectable at run time.

24. The system of claim 23, further comprising a filter configured to receive a notification when the local publisher publishes a message.

25. The system of claim 24, wherein the filter determines whether to forward the message or data derived from the message to one of the devices.

26. The system of claim 24, wherein the logger receives requests to log messages from an application and wherein the filter provides a notification when the application has completed.

27. The system of claim 24, wherein the filter is activated through a callback.

28. The system of claim 23, wherein the interface provides access to methods associated with the logger, the interface being customized to operate with a programming language or programming model.

29. The system of claim 28, wherein the programming model comprises a component object model (COM).

30. The system of claim 23, wherein the local publisher allocates a buffer when the local publisher is instantiated.

31. The system of claim 30, wherein the local publisher provides allocates memory from the buffer to receive the log message.

32. The system of claim 23, wherein the local publisher publishes a log message in an extensible markup language (XML) and places the published log message in a trace object.

33. The system of claim 23, further comprising an application configured to request that messages be logged via the logger.

34. The system of claim 33, wherein the application operates asynchronously with respect to the logger.

35. The system of claim 34, wherein the application continues executing even if there is insufficient memory to log the message.

36. The system of claim 33, wherein the application operates synchronously with respect to the logger.

37. The system of claim 33, wherein the set of one or more devices is selected after the application is compiled.

38. The system of claim 23, wherein each device is configured to transform a received log message for display, output, storage, or transmission.

39. The system of claim 23, further comprising a reader configured to read a trace comprised of data derived from the log messages and to display the trace in a hierarchical manner.

40. A computer-readable medium having computer-executable instructions, comprising:

selecting a set of one or more devices to receive log messages;

registering each device in the set with a publisher;

receiving a request to log a message; and

publishing, by the publisher, data derived from the message to the set of one or more devices.

41. The computer-readable medium of claim 40, wherein selecting the set of one or more devices to receive log messages comprises passing a parameter via a command line that is associated with an application configured to request logging of one or more messages.

42. The computer-readable medium of claim 40, wherein the parameter indicates that an indication of the set of one or more devices is in a database.

43. The computer-readable medium of claim 40, wherein selecting the set of one or more device to receive log messages comprises passing information via an environment variable.

44. The computer-readable medium of claim 40, wherein selecting the set of one or more devices to receive log messages comprises a combination of passing a parameter via a command line and passing information via an environment variable.

45. The computer-readable medium of claim 40, wherein publishing the message to the set of one or more devices comprises creating a trace object.

46. The computer-readable medium of claim 45, wherein the trace object is formatted in accordance with an extensible markup language (XML).

47. The computer-readable medium of claim 40, further comprising calling a filter to indicate that the message is available to be logged.

48. The computer-readable medium of claim 40, wherein publishing the data comprises providing a context identifier that identifies a context of a thread that sent the request and a context identifier that identifies the context of another thread associated with the thread.

49. The computer-readable medium of claim 40, wherein publishing the message comprises providing information that uniquely identifies the thread, the information comprising an identifier that identifies a machine on which the thread is

executing, a name of a process that spawned the thread, an identifier that identifies the process, and an identifier that identifies the thread.

50. The computer-readable medium of claim 40, wherein the data is published on a first machine and wherein the request to log a message is received from a second machine.

51. The computer-readable medium of claim 40, wherein registering each device in the set with a publisher comprises indicating what type of message or types of messages to publish to the device.